Image Retrieval in the Plenoptic Space

Alireza Ghasemi
AudioVisual Communications Laboratory
Ecole Polytechnique Fédérale de Lausanne (EPFL)
alireza.ghasemi@epfl.ch

***Abstract*—In this report we study the ways to exploit the vast amount of information inherent in the plenoptic space and constraints of the plenoptic function to improve the efficiency of image retrieval, recognition and matching techniques.**

The plenoptic space is formed by extending the notion of traditional two-dimensional by adding more dimensions for viewing direction, time and wavelength. Using current hand-held devices' built-in cameras, one can easily capture a large sequence of pictures from a single static scene by moving the camera in one direction, which form a three dimensional plenoptic function.

I. Introduction

The traditional notion of an image has two degrees of freedom, one for each of the image coordinates (x and y). In the context of digital signal processing, this translates to a two dimensional matrix.

However, we have the intuition that at least our eyes view in three dimensions. One may also argue that time is the fourth dimensions. Such evidences make us think about adding more degrees of freedom to the traditional notion of an image, which leads to the so-called "Plenoptic Function".

The most general form of plenoptic function contains 7 degrees of freedom (arguments). This is formed by adding 3 degrees of freedom for the spatial camera position, one for time and one more for the wavelength, to the tradition (x, y) arguments of the image plane. Therefore we have:

$$P = P_7(x, y, V_x, V_y, V_z, t, \lambda) \tag{1}$$

In which (V_x, V_y, V_z) is the spatial coordinate of the camera position.

The plenoptic function is highly structured and a significant amount of information about the captured scene can be extracted by exploiting this regularity. However, the general seven dimensional plenoptic function is extremely difficult to capture and operate. Therefore, we usually reduce the number of parameters by introducing constraints on the capture process. For example, we can consider only static scenes, thereby omitting the time index. Moreover we can omit the wavelength by considering single-sensor lenses. We may also enforce restrictions on the camera movements to reduce the number of parameters even more. Figure 1 depicts devices which are used to capture different subsets of dimensions of the plenoptic function.

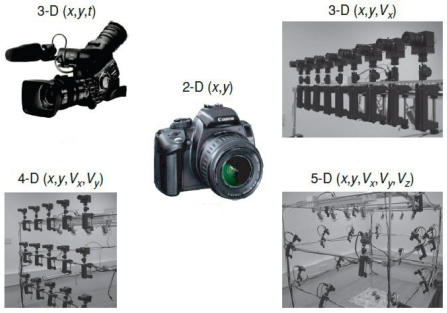


Figure 1: Capturing the Plenoptic Function in Different Dimensions

A special case of the plenoptic function has the special name of "the epipolar volume". Epipolar volumes are easily acquirable using traditional cameras yet contain a significant amount of useful information. We will discuss this special kind of plenoptic function in the next section.

A. Epipolar Volume

An epipolar volume is formed from a plenoptic function by restricting the camera motion to a straight horizontal line (and thereby omitting V_y and V_z from camera parameters), considering only static scenes (thereby omitting t) and also considering only one sensor (omitting λ). This 3 dimensional subspace of the plenoptic function is called the epipolar plane image or simply epipolar volume and is denoted as:

$$I = I(x, y, V_x) \tag{2}$$

Therefore, the epipolar volume is a three dimensional function with one dimension for the camera position along the X axis [1] and the other two for the image itself, i.e. horizontal and vertical axis of the image plane.

In the context of digital image processing, an epipolar volume is represented via a three dimensional matrix formed by stacking a sequence of images. Note that it may also be a

[1] or equivalently the time of taking the image, since camera speed is assumed constant. In fact, many literature paper denote the third dimension by t

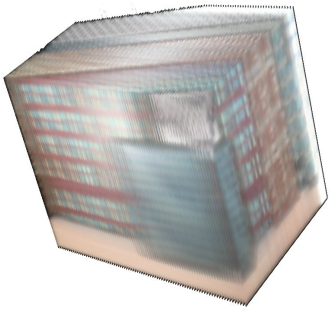


Figure 3: An Epipolar Volume

four dimensional matrix because of the three color channels (RGB) available for each image.

The epipolar volume is highly structured and follows regularity that can be exploited to extract information regarding the scene being captured. In the rest of this section we will study the structure of the epipolar volume.

Figure 3 shows an epipolar volume. Three different slices of the volume are visible in this picture.

B. Characteristics of the Epipolar Volume

We start our discussion on regularities of the plenoptic function by studying the properties of the epipolar volume using the pinhole camera model [7].

Consider taking a single picture using the pinhole camera model. Assuming that the focal length of the camera is unity and the optical center is located at the origin, the projection of a scene point (X, Y, Z) to the image plane is calculated as:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \mapsto \begin{pmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \end{pmatrix} \tag{3}$$

Now consider the case of an image sequence, taken by moving camera V_x units along the horizontal axis. Adding a third dimension for the camera location (set to V_x), now the mapping becomes:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \mapsto \begin{pmatrix} \frac{X}{Z} - \frac{V_x}{Z} \\ \frac{Y}{Z} \\ V_x \end{pmatrix} \tag{4}$$

This is how an epipolar volume is formed. We conclude the following two important facts regarding the equation (4).

1) Each scene points corresponds to a line (a single parameter curve) in one of the $x - V_x$ slices of the epipolar volume (the slice corresponding to $y = \frac{Y}{Z}$).

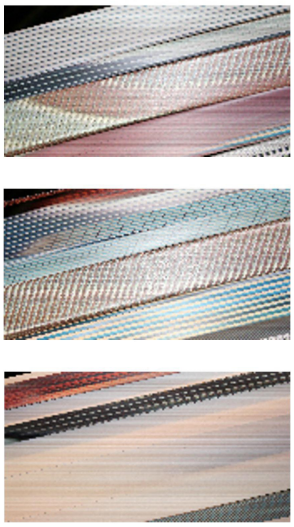


Figure 4: Three Epipolar Planes of an Epipolar Volume

2) The slope of this line is proportional to the depth (Z value) of the scene point. This means that lines corresponding to closer points are more horizontal.

Please note that the above facts are valid only when all our initial assumptions have been met. For example, if the motion is non-linear or speed varies, then the feature paths are no longer lines. An $x - V_x$ slice of an epipolar volume is called an "epipolar plane".

The two fact described above are the starting points for our analysis of the epipolar volume and its properties and how we use it to extract information about the scene.

The goal of this report is to study the properties of plenoptic function (especially epipolar image volumes) and propose efficient approaches to extract meaningful and discriminative features from such data using their coherent regularity. Our proposed method should as much as possible invariant to small changes in capturing environment and camera parameters and be adaptable to varying number of images in the sequence.

In the rest of this report we first review the definition of some important terms used throughout the text (section I-C). Then we review some related and literature (section II). After that we propose some approaches based on plenoptic information for feature extraction and study their properties (section III). Finally we illustrate some initial results of implementing the proposed algorithms (section IV).

C. Terminology

In this section we review the definition of the most important terms used throughout this report. These terms are:

- **Plenoptic Function:** Generalization of an image where as well as horizontal and vertical coordinates of the image we can vary the location of the camera in

(a) Caption

(b) Caption'

(c) Caption"

Figure 2: Three Sample Pictures of a Moving Image Sequence

space (3DOF in the most general form) and time and wavelength (The most general form).

- **Epipolar Image Volume:** A special of the general plenoptic function where the scene is static and camera can only move horizontally perpendicular to the image normal. It's a 3d function: $f(x, y, V_x)$ where V_x is the image index or the camera location.
- **Epipolar Plane:** A horizontal slice of the epipolar volume, i.e a function of the form $f(x, y_0, V_x)$ where y_0 is constant.
- **Epipolar Line:** A straight line segment in the epipolar plane. Epipolar lines are also called feature paths since each line corresponds to a unique scene point.

II. Related and Similar Works

The concept of plenoptic function and its properties was first introduced and discussed by Adelson and Bergen in [1]. Before them, [4] had proposed the concept of epipolar plane image (EPI) volume which is now known to be a special case of plenoptic function. They used the EPI volume for depth estimation.

In a more recent work, [3] introduced the concept of a plenoptic manifold. Plenoptic manifolds are trajectories traversed by scene points in the plenoptic space. Understanding plenoptic manifolds is crucial in studying the coherent regularity of plenoptic function. This regularity can be exploited in many different ways for processing and extracting information from images, as previously done by Bolles [4] for depth estimation.

The algorithm proposed in [2], exploits the regularity of the plenoptic function for more accurate image segmentation. It uses a modified version of the active contour algorithm which takes into account the depth information extracted from the plenoptic volume.

In [6] another work is presented which uses the plenoptic function for extracting information from images. The authors propose to extract layers (regions of image which lie in the same depth) of the image using the information contained in multiple views of the scene.

In [10], the regularity and redundancy of an epipolar image sequence is exploited to extract accurate feature correspondences between the first and the last image of the sequence. The paper uses a cost function based on the variance of intensity along candidate epipolar lines. Recall that in an epipolar plane, the intensity should not change along epipolar lines.

Wang et. al. proposed a multi-view object recognition algorithm based on properties of epipolar volumes [12]. In their approach, a disparity estimation method is utilized to improve the performance of single-view object recognition.

Another idea which is in some sense related to feature extraction from epipolar volumes is the concept of space-time interest points [8]. Space-time interest points are extensions of the concept of feature points (which are spatial) to both space and time dimensions. It means that space-time interest points are corners in space-time volumes corresponding to videos for example.

The main application of space-time interest points is in activity recognition from video sequences which is far from plenoptic image retrieval. However they bear some similarities with the field of plenoptic image processing. Both approaches are extension of simple two dimensional feature points and try to use these features for recognition. However, in space-time interest point detection usually no assumption is made regarding the camera motion and therefore information in epipolar planes is not utilized. A detailed explanation of space-time interest points can be found in [8].

III. The Proposed Feature Extraction Method

Recall from the introduction the importance of the epipolar planes in extracting information from three dimensional plenoptic function. We showed there that: 1) Each scene

point (i.e. distinguishable feature) maps to straight line in one of the epipolar planes in the plenoptic function which we call "feature path", and 2) The slope of a feature path is proportional to the depth of its corresponding feature point.

In this section we propose and study different way to exploit this information in obtaining high-quality feature points.

By high quality we mean that the feature points should be discriminative and repeatable. That is, the same set of feature points should be detected under illumination, scale and orientation changes and the features must be discriminative enough to distinguish between different scenes.

A. Detection and Extraction of Lines

Now, we know that the first step in any feature extraction method for epipolar volumes should be detection and extraction of line segments in epipolar planes.

The most common method for line extraction from images is the Hough transform [11]. In the Hough Transform, a line is parametrized as the set of (x, y) points satisfying the following equation:

$$x\cos\theta + y\sin\theta = \rho \tag{5}$$

We can see that in this model that the set of parameters (ρ and θ) is different from those used the traditional representation of two dimensional lines (slope and intercept). The reason for using this parametrization is that using the traditional model $y = mx + h$, the slope m is not bounded whereas θ in Hough model is bounded to $-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$.

Having this model we now define the new $\rho-\theta$ space. The duality between the original $x-t$ space [2] and the transformed $\rho - \theta$ space is such that each point in the $x - y$ plane is mapped to a harmonic curve in the $\rho - \theta$ plane. Moreover, each point in the $\rho - \theta$ space corresponds to a unique line in the $x - y$ plane.

The Hough Transform works by constructing a $\rho - \theta$ plane of the candidate lines of the $x - y$ plane. Usually an edge detection algorithm is applied to the image as the preprocessing step. There are two reasons for this. First reason is that Hough Transforms works on binary images. Therefore some kind of thresholding is essential to apply to the image. The second reason is that edge detection helps in removing irrelevant points since line usually lie in th edges of the image.

As well as edge detection and before that, a low-pass filter is also appropriate to apply to the image. The reason for this is to reduce noise before edge detection is applied so that it doesn't detect noisy points as edges.

Figure 5 shows pseudo-code of the Hough Transform algorithm.

[2]Remember that Hough Transform is applied to epipolar planes which are in the $x - t$ space

Require: The two dimension matrix I representing an image.
1: Apply a low-pass filter to I.
2: Apply an edge detection algorithm to I and convert it to a binary image.
3: Discretize the range of θ values in the vector Θ.
4: Discretize the ρ parameter into n_ρ distinct values.
5: Construct the $Length(\Theta) \times n_\rho$ output matrix H.
6: Set all elements of H initially to 0.
7: **for** each feature point (x, y) in I **do**
8: **for** each $\theta \in \Theta$ **do**
9: $\rho = x\cos\theta + y\sin\theta$
10: $H(\theta, \rho) = H(\theta, \rho) + 1$
11: **end for**
12: **end for**
13: **return** the output image H

Figure 5: The Hough Transform Algorithm

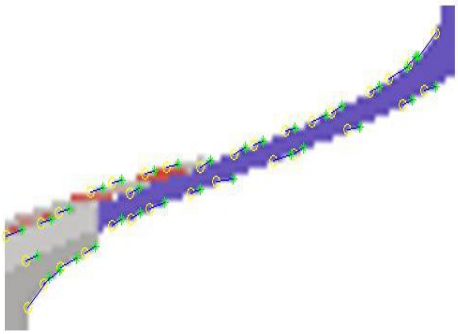


Figure 6: Lines Detected by the Hough Transform on an Epipolar Plane

A set of collinear points in the $x - y$ plane correspond to intersection of their corresponding curves in the $\rho - \theta$ curve which means more intensity in the points of intersection. Therefore, prospective lines correspond to $\rho - \theta$ points with highest intensity (i.e. peaks) and we should select points with intensity higher than a threshold τ_1.

After selecting peaks from the $\rho-\theta$ plane, we can form the lines in the $x - y$ plane by determining their two endpoints. In this step we can prune the set of detected lines and only retain the most stable ones. For example we can omit the lines shorter than a threshold τ_2. Figure 6 depicts the visualized results of running the Hough Transform on an epipolar plane.

1) Improving Line Detection Using Characteristics of Epipolar Planes: The Hough transform is the most common way to detect arbitrary lines in an image. However, due to the increased number of parameters and noise and textures of the images, some erroneous results are also returned by the algorithm.

As well as the problem of erroneous lines, Hough trans-

form also faces the problem of discretizing values of ρ and θ which causes the algorithm to run quite slowly. We can use the prior information on characteristics of epipolar planes to improve both the accuracy and time complexity of the Hough transform.

To improve runtime complexity of the Hough transform we notice the fact that, by assuming camera movement in a single direction, all lines in all epipolar planes will have either positive or negative slope, and not both. If the camera moves from left to right, all epipolar lines will have positive slope and if the camera moves from right to left, all epipolar lines will have negative slope.

We can use this fact to compress the search space for θ and thereby time complexity of the Hough transform. We don't need to discretize and search the whole range of θ ($-\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2}$). We have to just search half of it (either $-\frac{\pi}{2} \leq \theta \leq 0$ or $0 \leq \theta \leq \frac{\pi}{2}$). This reduces running time of the algorithm by a factor of 2.

To further reduce the running time of the algorithm, we note the fact that Hough Transform's runtime complexity is proportional to the number of feature points (in a binary image). Applying a low-pass filter to the image before thresholding and edge detection reduces the number of "on" points in the output image. Moreover, low-pass filter has the extra effect of reducing the effect of textures and noise in computing the hough transform.

To improve accuracy, we note that each epipolar line (i.e. a feature path) corresponds to a single unique feature. It means that all point along an epipolar line should ideally have equal intensity (or color). Therefore we can do a post-processing step after the hough transform and ignore lines for which variance of intensity along the points is above some threshold $\sigma_{intensity}$. More complicated post-processing steps may similarly apply.

B. Selection of Feature Points

After applying Hough Transform to all epipolar planes, we have the epipolar line segments which are paths traversed by image features. Now we can use these lines to extract visual features from the epipolar volume.

As noted earlier, each epipolar line segment corresponds to a single unique image features. So we can simply use the starting point of each line segment as a feature point. Considering the epipolar plane at which the line resides, we can compute the full three dimensional coordinates of the feature point ((x, y, V_x)) and the specific image (V_x index) in which it lies. These information will be used to describe the point as a feature vector.

Note that all points of a line segment represent the same feature. Therefore it does not actually matter which point (along the line segment) is selected as the feature points. We can equivalently select the endpoint or middle point of the line segment as well.

Require: The input epipolar image volume V.
1: Set $S = \varnothing$.
2: **for** each epipolar pplane P in V corresponding to $Y = y_0$ **do**
3: Apply a low-pass filter to P.
4: Apply edge detection to P and convert it to a binary image.
5: Apply the Hough Transform to P and extract all line segments.
6: **for** each starting point (x_i, V_{xi}) of a line segment **do**
7: Select (x_i, V_{xi}) as a feature points.
8: Add point (x_i, y_0, V_{xi}) to the set of features S
9: **end for**
10: **end for**
11: **return** the set of feature points S

Figure 7: The Proposed Feature Detection method

Applying the same extraction to all epipolar planes and aggregating the results, we collect the complete set of feature points describing the whole epipolar volume.

Using this approach we avoid correspondence estimation between images of the volume. We also don't need to perform general feature tracking since we are using the prior information about the camera motion to efficiently recover feature paths. Therefore we are using all available information in extracting this set of feature points.

Th size of the feature set is significantly smaller than the case when we simply perform traditional feature extraction on all images of the volume and aggregate the results. This is because of the fact that many redundant features (points along the same epipolar line) are ignored. A second reason would be that many unstable features (those that appear only on a few images) are not selected using our proposed approach since we omit lines shorter than a threshold. A feature should be present in (at least) more than one image to be selected by our approach.

Figure 7 depicts the pseudo code for the proposed feature detection method.

C. Invariance to Scaling, Rotation and Illumination Changes

An efficient feature detector should be invariant to transforms in the scene. It means that the algorithm should detect and extract the same feature points even after applying some (at least) simple transforms to the image, so that recognition and correspondence becomes possible.

In this section we will intuitively show that our feature detection approach is invariant to simple affine transforms that happen quite frequently in real-world capturing scenarios.

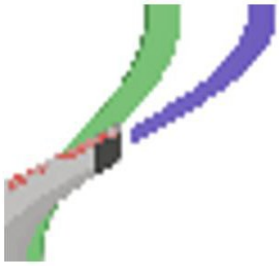
Three of the most common transforms occurring in natural pictures are scaling, rotation and illumination change. We start our discussion by arguing about how transforms in captured images change corresponding epipolar planes.

(a) Original Epipolar Plane

(b) The Same Epipolar Plane in the Scaled Volume

Figure 8: Effect of Scaling on the Epipolar Planes

(a) Original Plane



(b) Plane in the Rotated Iamge

Figure 9: Effect of Rotation on the Epipolar Planes

1) Scaling: Scaling an image can be described by moving the camera further from or closer to the scene, i.e. moving the camera perpendicular to the motion axis. Therefore we can easily conclude that scaling is equivalent to increasing the depth of all scene points. As we already showed, depth of feature points is proportional to their corresponding feature path (epipolar line).

Summing the two facts stated above, we can conclude that scaling the in the image plane corresponds to a kind of rotation in the epipolar planes. It is not a simple rotation since slope of the lines can not exceed infinity (fully vertical). Therefore, more distance object (those with larger Z value) rotate less than closer objects. Figure 8 depicts this phenomenon.

Moreover, we know that Hough Transform is rotation invariant. A rotated line segment is still a line segment and can be detected by the Hough Transform.

Aggregating the facts stated above we can conclude the final lemma as below:

Lemma 1: The proposed feature extraction method is invariant to scaling the image plane. It means that after moving the camera further from or closer to the scene, the same features are still extracted using the proposed algorithm (excluding those points that disappear due to decreased resolution)

2) Rotation: Rotation can not be described in the epipolar plane as easily as scaling. However we try to describe it using simpler basic transforms to justify about invariance of our algorithm to rotation.

When the same scene is captured with two or more different camera orientations (i.e. one is rotated with respect to the other in the image plane) what happens in the epipolar plane is that some point are delivered from their original epipolar plane (i.e. Y index) to another epipolar plane and merge with original points in that plane. If no occlusion occurs in the new plane, then we can expect that those points form the same feature path, but in the new plane. Figure 9 shows a typical effect of rotating the camera on the epipolar plane for a simple scene.

According to the evidence stated above and remembering the fact that extraction algorithm is executed on all epipolar planes, we can conclude the following lemma:

Lemma 2: The proposed feature extraction method is invariant to rotation of the image plane. It means that after rotating the camera , the same features are still extracted using the proposed algorithm (excluding those points that disappear due to occlusion with original point of the destination plane)

3) Invariance to Illumination Changes: It is easier to describe the effect of illumination changes on the epipolar since the effect is quite minor. Change of illumination of the scene is equivalent to changes of intensity of points in the epipolar planes. However, since an edge detection algorithm is applied to the planes before Hough Transform and the only important factor in determining edges is the relative intensity of scene points, we can conclude the following lemma:

(a) Original Epipolar Plane

(b) The Same Epipolar Plane with Illumination Changed

Figure 10: Effect of Scaling on the Epipolar Planes

Lemma 1: The proposed feature extraction method is invariant to illumination changes in the image plane, provided that the modified illumination is applied uniformly to all of the scene and all captured images. It means that after changing the light source, its position or its power, the same features are still extracted using the proposed algorithm (excluding those points that disappear due lack of brightness)

Figure 10 shows how changing illumination affects epipolar planes.

D. Alternative Approaches

Extracting feature points directly from the output of the Hough Transform has the benefit of utilizing the information of the epipolar planes. However, the set of selected points is highly dependent on the edge detection output and also the traditional criteria for feature detection (corner,...) are not explicitly addressed (although they are usually met). In this section we will have a look at other possible ways for feature extraction in the plenoptic space.

1) The Sliding Windows Approach: Instead of using starting point of each epipolar line as the feature point, we can use information about the local distribution of lines and their orientations in different parts of the epipolar planes to find more robust feature points.

To accomplish this goal, we consider a window of fixed size w around each prospective feature point. We find all epipolar lines which fully or partially lie inside this window. The number of lines and the distribution of their orientations

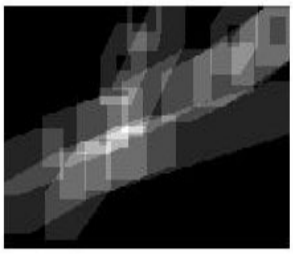


Figure 11: Significance of Points in the Sliding Window Approach

is a good measure of how significant a prospective feature point is. We call this measure the "Significance Ratio".

The significance ratio measures the number of epipolar lines passing through neighborhood of a point. The more lines passing from neighborhood of a point, the more significant the point is and more probably it will be a feature point. As well as the number of lines, we may also consider their orientations. The windows in which there are lines with many different orientations can be considered more significant than those with many equal-orientation lines in neighborhood. Variance of orientations (or slopes) of lines could be considered a good measure of diversity. By considering line variances, we are implicitly assuming that feature points lie in area in which depth changes. This is usually true.

In the most general case, we can compute the significance ratio for all points in all planes and select as features those points whose significance ratio exceeds a predefined threshold. Figure 11 shows the significance map for an epipolar plane. Brighter points have higher significance.

2) Two-Dimensional Corner Detection As Preprocessing: The sliding window approach described above runs significantly slow, especially with a naive implementation. This is mostly because of the fact that in the most general case, significance ratio should be computed for all points of plane and maximal points get selected as features.

To improve running time of the sliding windows approach we should reduce the number of points for which significance ratio is calculated. It means that we have to obtain an initial set of candidate feature points and then prune the set using the significance ratio.

The obvious way for obtaining the initial feature set would be using a traditional two-dimensional feature detection algorithm as the pre-processing step to obtain a reduced set of feature points. Then we can compute the significance ration on these points and retain only those points for which the ratio exceeds a threshold $r_{threshold}$. However, this approach does not consider correspondences between points is therefore vulnerable to redundancy.

3) The Hybrid Two-Step Approach: The modified sliding window approach described above has the main advantage

that it uses the plenoptic function to prune the set of selected feature points. However, the runtime complexity is still relatively high and moreover, the sliding window approach does not explicitly consider the problem of correspondent points in different image planes. Multiple image points that are mappings of the same scene point should not be retained in the final feature set since they are highly redundant. In this section we propose an improved hybrid approach which addresses these problems.

Our proposed approach is a two step method taking into account both the image plane and epipolar plane information in finding feature points.

In the first step, we apply a two-dimensional corner/feature detection approach to all of the image planes and extract a relatively large set of prospective feature points.

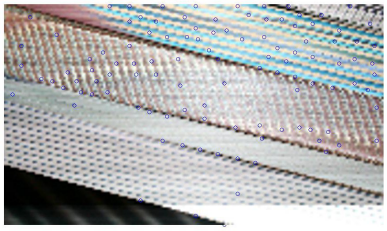
In the second step, we move to the epipolar plane space and extract the lines formed (only) by the prospective points of the previous step. To achieve this goal, first we repartition the points from consecutive image planes to consecutive epipolar planes. Then, we form a binary image in place of each epipolar plane by considering a bright point ("1") for points which correspond to a prospective feature.

We apply the Hough Transform to this synthetic epipolar plane and extract the lines using the transform space. Finally, we retain only the starting point of each line and omit other points, in oder to minimize redundancy.

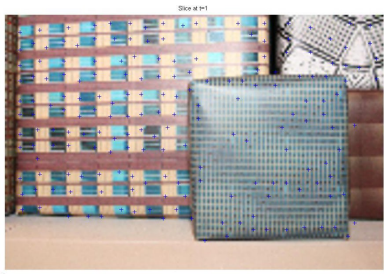
Our new approach has some advantages over the methods proposed earlier. First of all, it uses information in both image space and the epipolar plane space. Therefore, this approach is more accurate in determining corners and other features of the scene.

The second advantage is that it uses plenoptic information in determining correspondence between points in consecutive images and thereby avoids redundancy in the final feature set.

Moreover, in the two-step approach Hough transform can be made significantly faster since it is not executed on the whole epipolar plane but only on a selected subset of its points. Since runtime complexity of the Hough Transform is dependent on the number of bright points in the input binary image, the overall running time is significantly reduced. Figure 12 depicts sample feature points detected by the two step approach in the image and epipolar plane space.



(a) Epipolar Plane



(b) Image Plane

Figure 12: Two-Step Point Detection in the Image Plane and the Epipolar Plane

E. Matching Between a Single Image and an Image Sequence

So far, we have considered only the problem of extracting and describing feature points of a three-dimensional epipolar plane. We have assumed so far that in both training and testing phase, we have access to a sequence of images taken using a moving camera from a static scene (epipolar plane volume settings).

However, in a real-world scenario we can't usually assume a full plenoptic image as the test sample. Instead, a single image or a much shorter sequence is given as the input test sample for which we should find the corresponding scene. In this section we will see how this can be approached using our proposed methods.

The fundamental prerequisite to be able to do matching between a single image and an image sequence, is that equivalent or similar types of features get extracted from both data types (image and sequence).

This is the case in our two-step method proposed above. In the two-step method, we extract a set of two-dimensional feature points from all images in the sequence, and then remove unstable and redundant points using the regularity constraints of the plenoptic function.

This process can be easily adapted to the case of varying number of images in the sequences, and the extreme case of a single image as well. In a single image the epipolar planes are not available. Therefore, the first step becomes more dominant and less or no redundancy removal is performed. However the whole process is still executable. Since the type of extracted features is exactly the same, we expect that the correct scene is found by matching features of the image against features of the sequence.

F. Feature Description

After detecting feature points, we should describe them using multi-dimensional feature vectors. This is done using a feature description method. There are various feature description algorithms, generating vectors of various dimensionality. An appropriate feature description should be as short as possible while retaining distinguishability between different features.

To obtain a feature vector, the feature point itself as well as its neighboring points are considered and based on intensity of these points, a feature vector is extracted. The extracted vector should be as much as possible invariant to affine transformations in the image.

Various feature descriptors have been proposed in the literature so far. Among them, SIFT[9] has shown good performance. However SIFT is unacceptably slow, especially for real-time applications. In recent years, a simpler and faster feature descriptor has been proposed called BRIEF[5] which generates a binary vector, instead of SIFT's real vector. BRIEF is not as accurate as SIFT but its significance speed improvement makes it a choice for real-time applications.

For our proposed plenoptic feature extraction methods, there are two possible choices. One choice is to consider a three-dimensional (x,y and V_x) neighborhood of feature points and therefore a three-dimensional feature descriptor. For example, BRIEF can be easily extended to three dimensions.

The other choice is to use a traditional two dimensional descriptor by considering only the image plane in which the feature point falls. This is reasonable since feature paths all represent a single unique feature and therefore it may be redundant to consider multiple image planes to extract a feature descriptor.

Using a two dimensional descriptor has the extra advantage that it makes it easier to perform matching between volumes of different thicknesses and even matching between a volume and a single image, as explained in the previous section.

IV. Experiments

In this section we will demonstrate some of the experimental results obtained by the proposed feature extraction methods.

A. The Environment

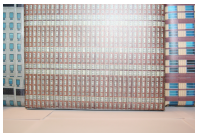
We implemented the algorithms in MATLAB. For testing algorithm we collected both synthetic and real datasets.

For capturing a real scene, we used a precise plenoptic device which is able to capture four dimensional plenoptic function of a scene by allowing accurate two-dimensional motion control of the camera. We constructed a physical model of an urban scene with miniature buildings of various sizes. The reason for using an urban scene is that the urban image recognition will be principal testing environment for our system.

For the synthetic model, we used the Blender rendering software. We created a simple scene with a single building-like cube structure and a simple texture, so that we could easily see the effects of changing the light source and camera location and parameters.

(a)

(b)

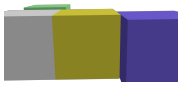


(c)

(d)

Figure 13: The Constructed Physical Model

(a)



(b)

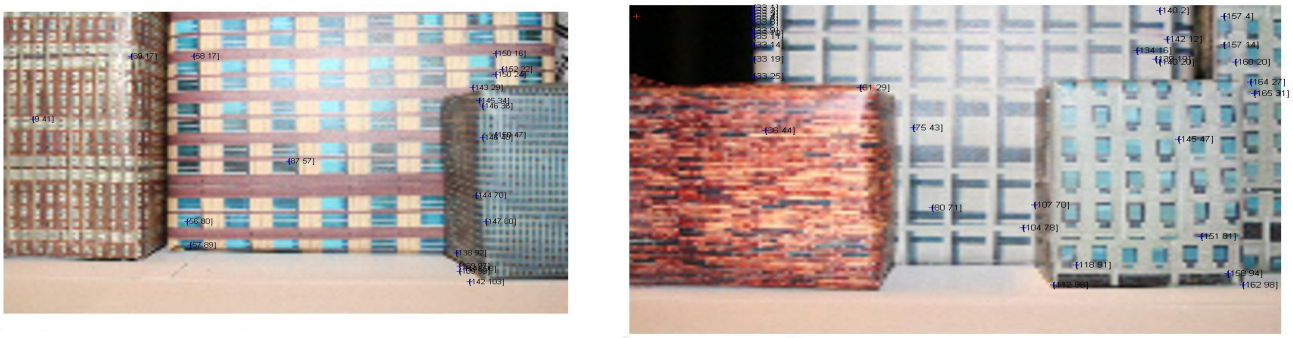
(c)

Figure 14: The Virtual Models Created Using Blender

B. Performance of Feature Detection

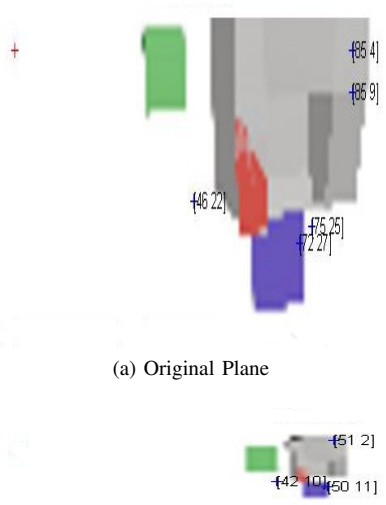
Figure 15 shows the detected feature points on two images of an epipolar volume. We have used the method of selecting starting points of line segments here. We can notice that most corner points have been detected, with other corners probably being detected in neighboring image planes.

Also, as an example of how affine transformations affect the process of feature detection, notice the figure 16 which shows detected points before and after scaling the scene. We can see that most of the points have been repeated in the scaled scene, despite those points that have disappeared due



(a) (b)

Figure 15: Detected Features on Two Samples of the Epipolar Volume of the Physical Model



(a) Original Plane

(b) Scaled Plane

Figure 16: Effect of Scaling on the Detected Points

to decreased resolution. Again we have used the method of selecting starting points of line segments.

V. Conclusion and Future Works

In this report, we proposed and discussed a feature extraction algorithm which is based on exploiting the information contained in the plenoptic function and redundancy of multi-view images.

For future improvements of the algorithm, we can exploit the correlation between subsequent epipolar planes to reduce running time of the Hough transform. Moreover, we can reduce the set of extracted features by enforcing a minimum distance between every two extracted feature points.

More preprocessing and post-processing steps can also be used to improve the algorithm. For example we may apply non-maximal suppression to the extracted points.

References

[1] E. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," *Computational models of visual processing*, vol. 1, pp. 3–20, 1991.

[2] J. Berent and P. Dragotti, "Segmentation of epipolar-plane image volumes with occlusion and disocclusion competition," in *Multimedia Signal Processing, 2006 IEEE 8th Workshop on*. IEEE, 2006, pp. 182–185.

[3] ——, "Plenoptic manifolds," *Signal Processing Magazine, IEEE*, vol. 24, no. 6, pp. 34–44, 2007.

[4] R. Bolles, H. Baker, and D. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision*, vol. 1, no. 1, pp. 7–55, 1987.

[5] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," *Computer Vision–ECCV 2010*, pp. 778–792, 2010.

[6] A. Criminisi, S. Kang, R. Swaminathan, R. Szeliski, and P. Anandan, "Extracting layers and analyzing their specular properties using epipolar-plane-image analysis," *Computer vision and image understanding*, vol. 97, no. 1, pp. 51–85, 2005.

[7] R. Hartley, A. Zisserman, and I. ebrary, *Multiple view geometry in computer vision*. Cambridge Univ Press, 2003, vol. 2, no. 00.

[8] I. Laptev, "On space-time interest points," *International Journal of Computer Vision*, vol. 64, no. 2, pp. 107–123, 2005.

[9] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[10] M. Matousek and V. Hlavác, "Epipolar plane images as a tool to seek correspondences in a dense sequence," in *CAIP*, ser. Lecture Notes in Computer Science, N. Petkov and M. A. Westenberg, Eds., vol. 2756. Springer, 2003, pp. 74–81.

[11] L. Shapiro and G. Stockman, "Computer vision. 2001," 2001.

[12] Y. Wang, M. Brookes, and P. Dragotti, "Object recognition using multi-view imaging," in *Signal Processing, 2008. ICSP 2008. 9th International Conference on*. IEEE, 2008, pp. 810–813.